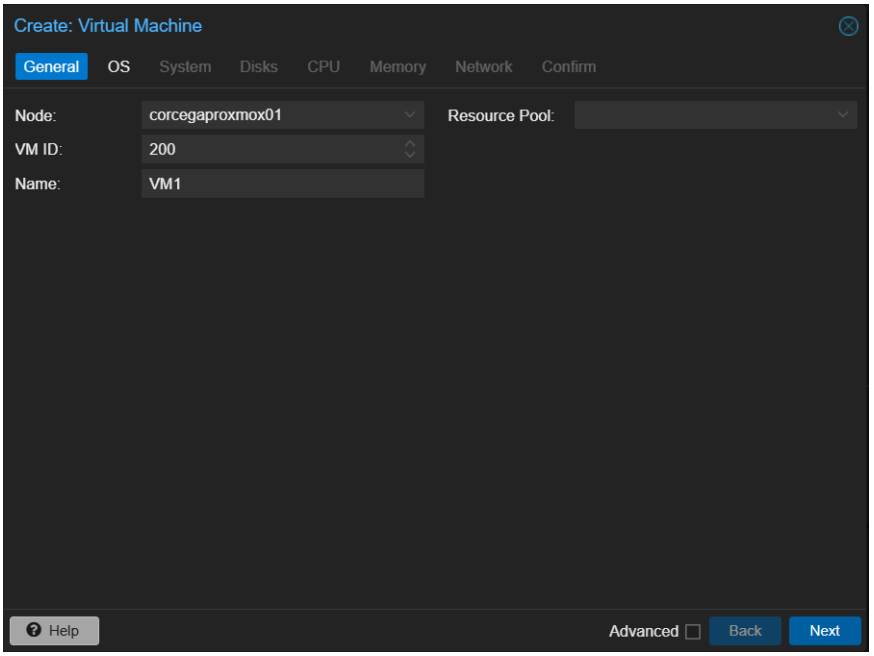
INTRODUCCIÓN A PROXMOX

Recursos disponibles:
(Medidas de uso actual sin uso adicional)

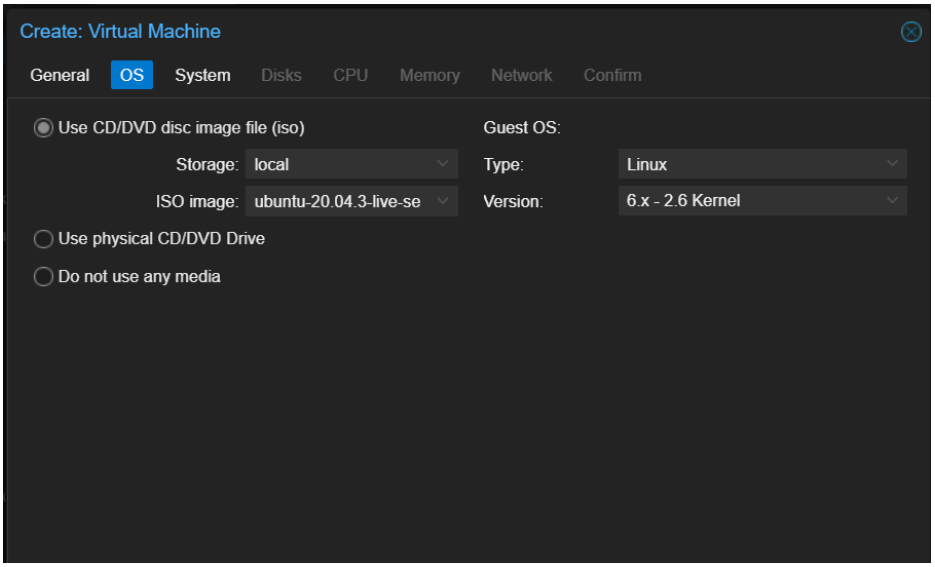
- **CPU:** 12 núcleos -> usa entre 0.10% y 0.20%
- **RAM:** 64 GB -> 3.61% usado 2.23 gib
- **Almacenamiento:** 1TB ->28.51% usado 26.78 gib
- **Redes:** enp3s0, vmbr0

Creación de una máquina virtual dentro del nodo:

Primero añadimos una id concreta para tener así las máquinas organizadas a futuro y un nombre para tenerlas reconocidas.

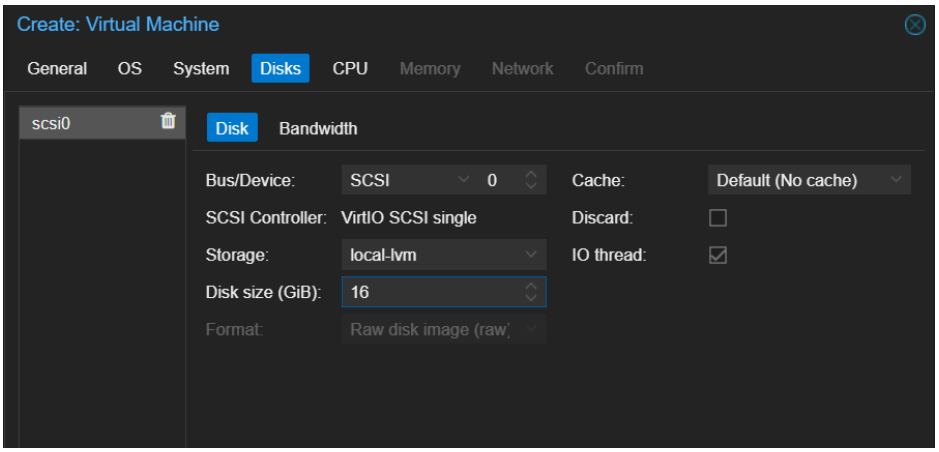


Seleccionaremos la ISO escogida en este caso un ubuntu

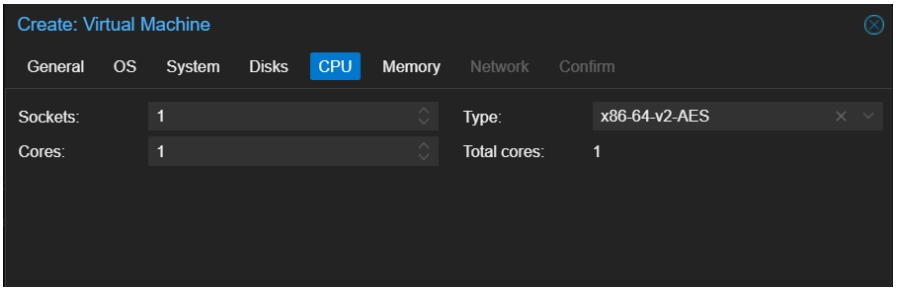


Pol Chacón, David Fernández y Sergi Estrella

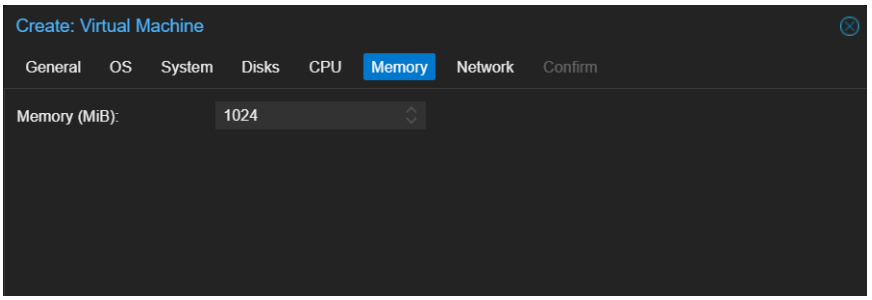
Podemos modificar la capacidad de disco de la máquina la dejamos en 16 GiB



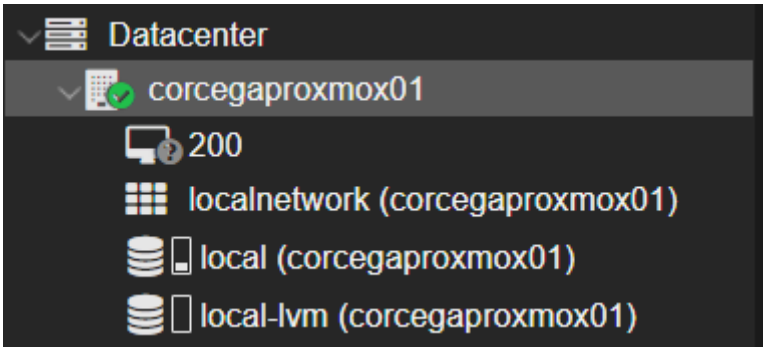
Aquí escogeremos la cantidad de sockets y núcleos que usaremos



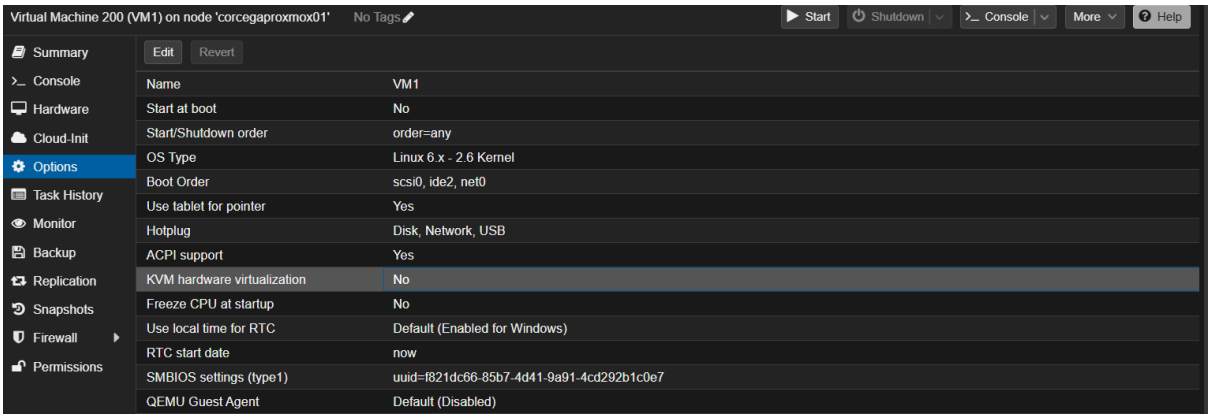
Aquí marcamos la memoria en mb. En nuestro caso solo un Giga.



Vemos que la máquina ha sido creada con éxito.

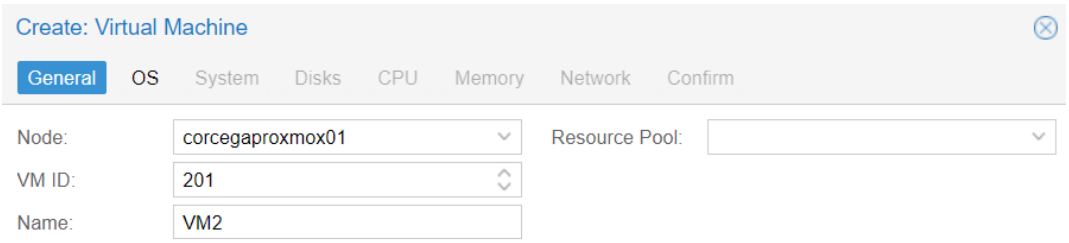


Y por último desactivamos una opción para poder encender y usar la máquina en cuestión.

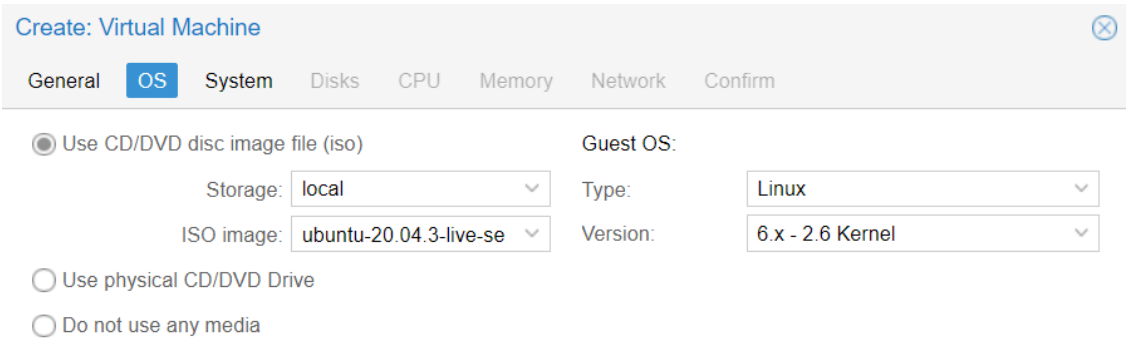


En los siguientes pasos recreamos lo hecho en la máquina anterior.

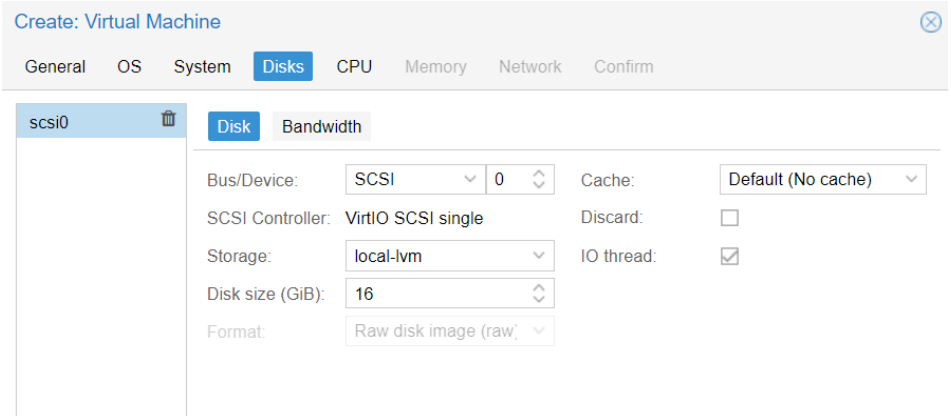
Primero añadimos de nuevo una id concreta en esta máquina será la 201.



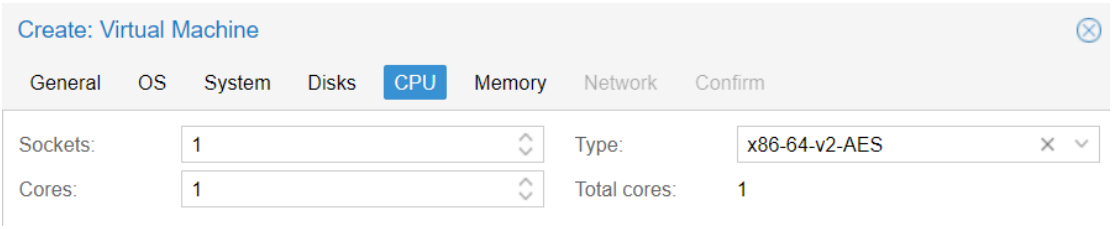
Seleccionaremos la misma ISO que antes la cual es la de ubuntu.



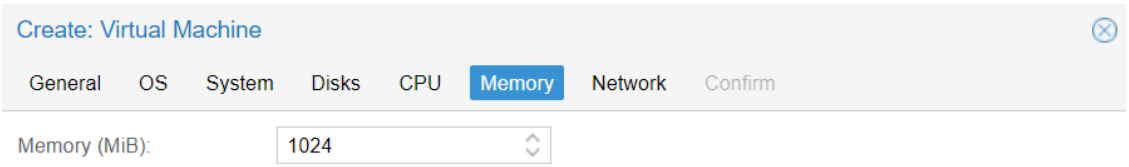
Podemos modificar la capacidad de disco de la máquina la dejamos en 16 GiB como la anterior.



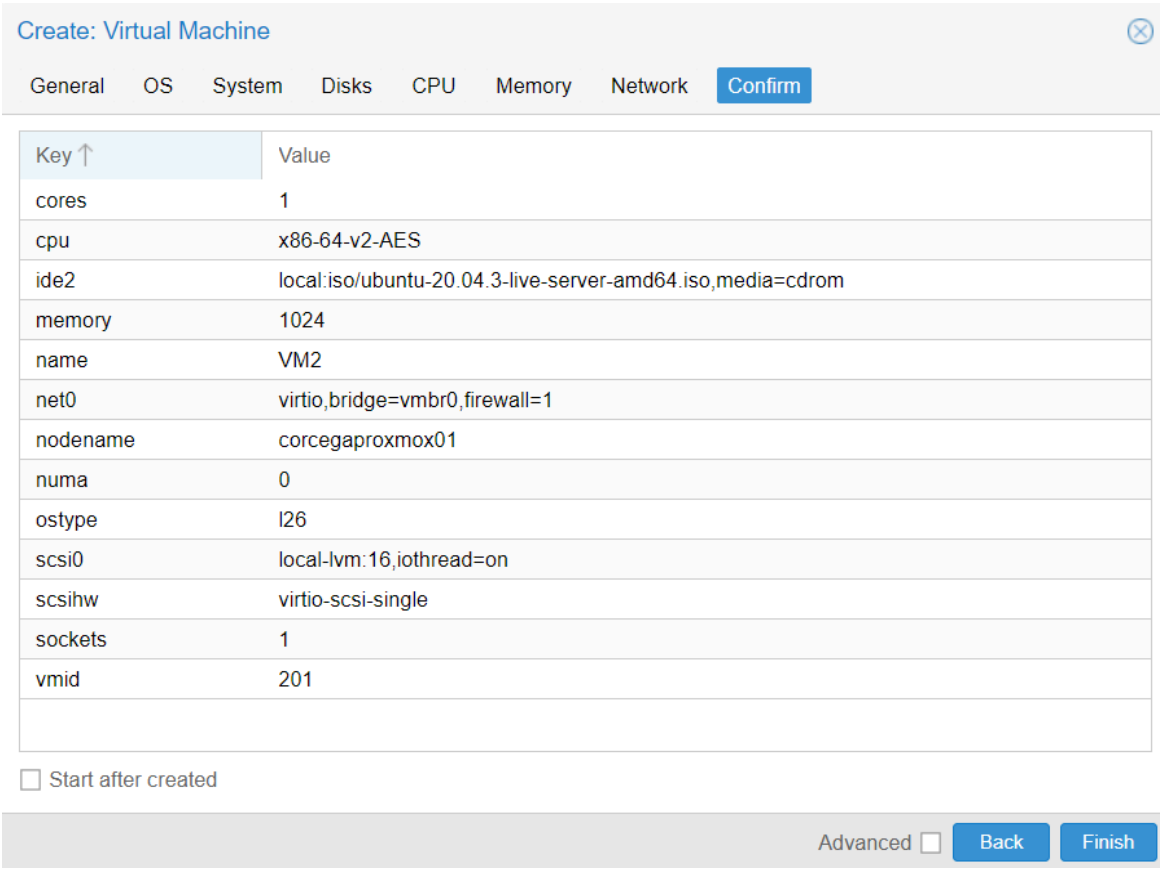
Aquí escogeremos la cantidad de sockets y núcleos que usaremos.



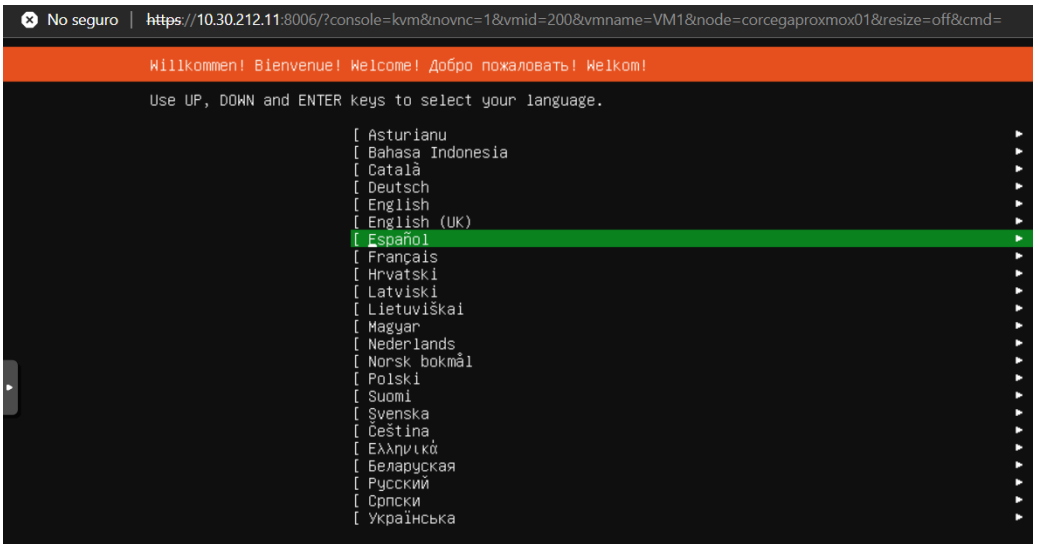
Aquí marcamos la memoria en mb. En nuestro caso solo un Giga.



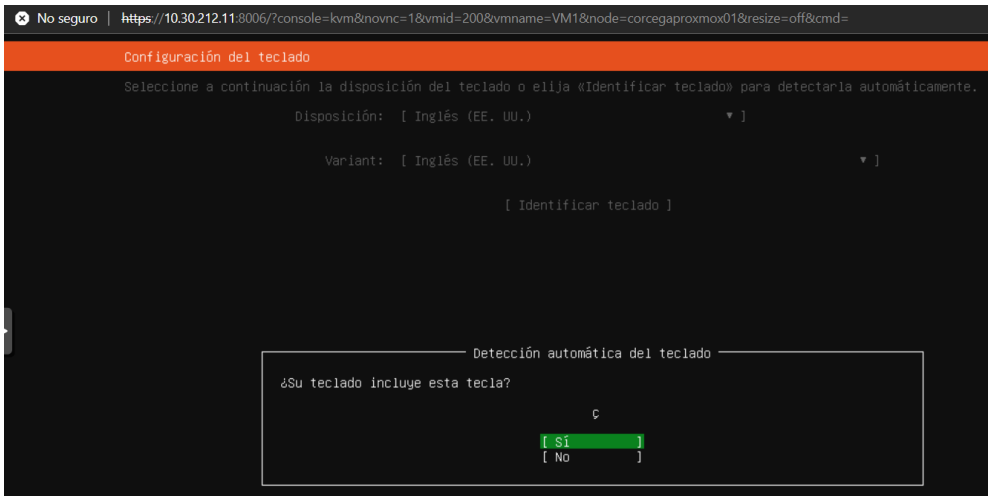
Y por último desactivamos la misma opción que antes para usar la segunda máquina.



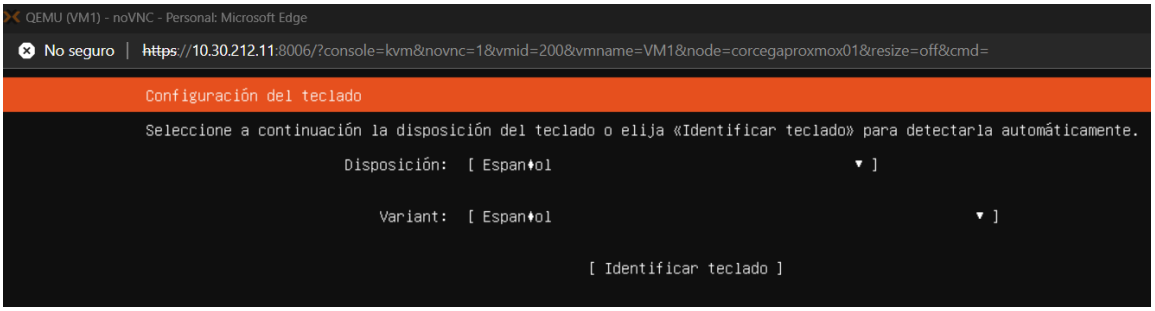
Seleccionamos el idioma que usará la máquina virtual.



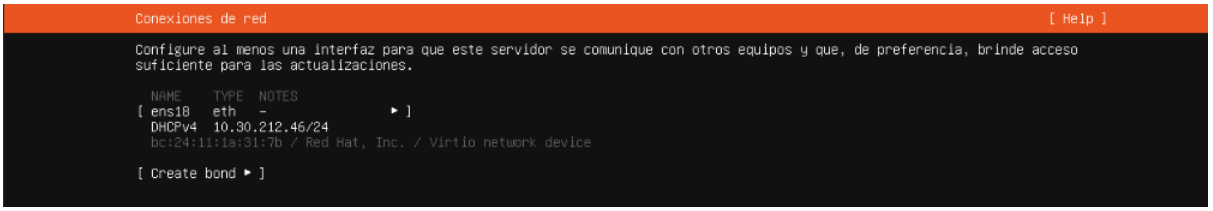
Realizamos la identificación de teclado al español.



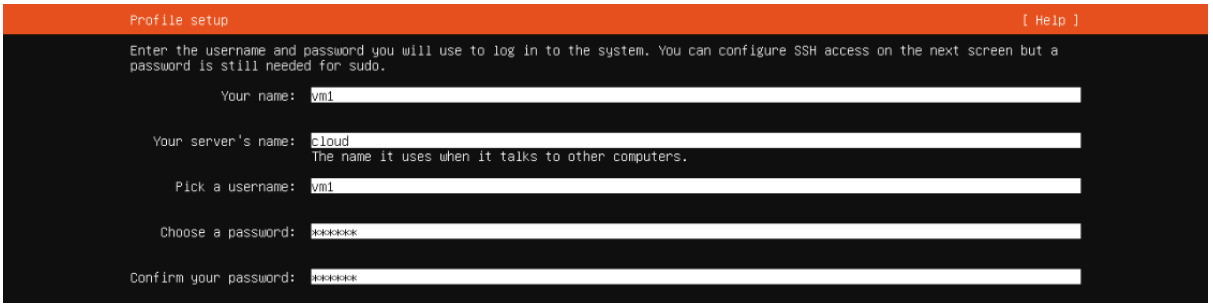
Aquí se ve que la identificación de teclado ha funcionado correctamente.



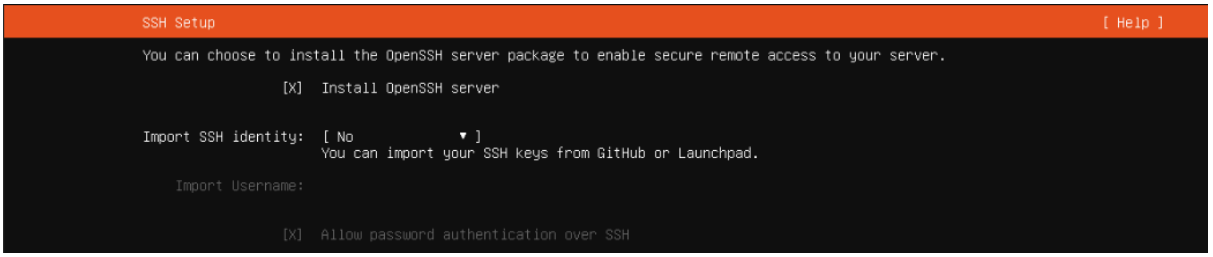
Aquí vemos la red de la máquina.



Escogemos el nombre de usuario y servidor junto a la contraseña.

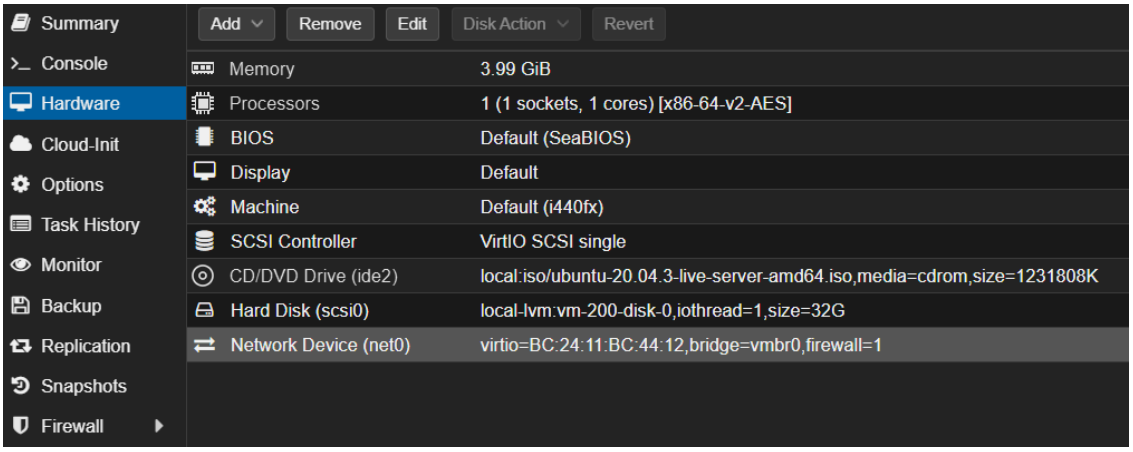


También instalamos el ssh para poder conectarnos a la máquina en un futuro.

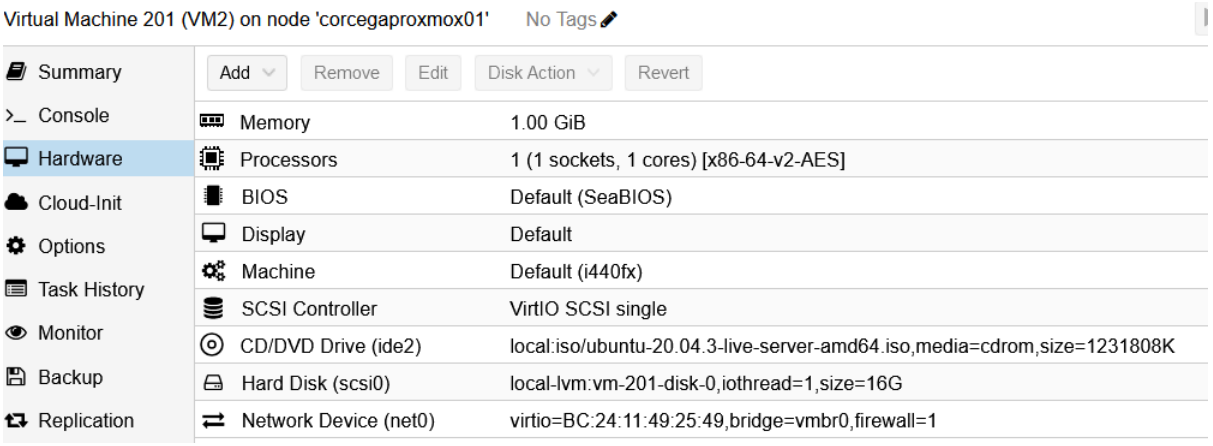


3. Configuración de Redes

Aquí vemos el adaptador de red en bridge.



Lo mismo que en la máquina anterior pero en la segunda máquina.



Hacemos ping de la maquina vm1 a 8.8.8.8 y a www.google.es

```
vm@cloud:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=116 time=12.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=10.6 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=116 time=10.5 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=116 time=10.5 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 10.460/11.034/12.616/0.914 ms
vm@cloud:~$ ping www.google.es
PING www.google.es (216.58.215.131) 56(84) bytes of data.
64 bytes from mad41s04-in-f3.1e100.net (216.58.215.131): icmp_seq=1 ttl=116 time=12.1 ms
64 bytes from mad41s04-in-f3.1e100.net (216.58.215.131): icmp_seq=2 ttl=116 time=11.8 ms
64 bytes from mad41s04-in-f3.1e100.net (216.58.215.131): icmp_seq=3 ttl=116 time=11.2 ms
64 bytes from mad41s04-in-f3.1e100.net (216.58.215.131): icmp_seq=4 ttl=116 time=11.2 ms
^C
--- www.google.es ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 11.243/11.603/12.114/0.373 ms
```

Aquí hacemos lo mismo de la vm2 a 8.8.8.8 y www.google.com.

```
vm2@cloud:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=116 time=11.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=10.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=116 time=9.24 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=116 time=10.6 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 9.242/10.604/11.843/0.922 ms
vm2@cloud:~$ ping www.google.com
PING www.google.com (216.58.215.132) 56(84) bytes of data.
64 bytes from mad41s04-in-f4.1e100.net (216.58.215.132): icmp_seq=1 ttl=116 time=12.3 ms
64 bytes from mad41s04-in-f4.1e100.net (216.58.215.132): icmp_seq=2 ttl=116 time=11.8 ms
64 bytes from mad41s04-in-f4.1e100.net (216.58.215.132): icmp_seq=3 ttl=116 time=11.7 ms
64 bytes from mad41s04-in-f4.1e100.net (216.58.215.132): icmp_seq=4 ttl=116 time=10.4 ms
^C
--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 10.366/11.546/12.342/0.724 ms
```

Aquí vemos la ip de la vm1.

```
vm@cloud:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:bc:44:12 brd ff:ff:ff:ff:ff:ff
    inet 10.30.212.50/24 brd 10.30.212.255 scope global dynamic ens18
       valid_lft 604631sec preferred_lft 604631sec
    inet6 fe80::be24:11ff:febc:4412/64 scope link
       valid_lft forever preferred_lft forever
```

Aquí vemos la ip de la vm2.

```
vm2@cloud:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:49:25:49 brd ff:ff:ff:ff:ff:ff
    inet 10.30.212.41/24 brd 10.30.212.255 scope global dynamic ens18
       valid_lft 553933sec preferred_lft 553933sec
    inet6 fe80::be24:11ff:fe49:2549/64 scope link
       valid_lft forever preferred_lft forever
```

Ping de la máquina vm1 a la vm2.

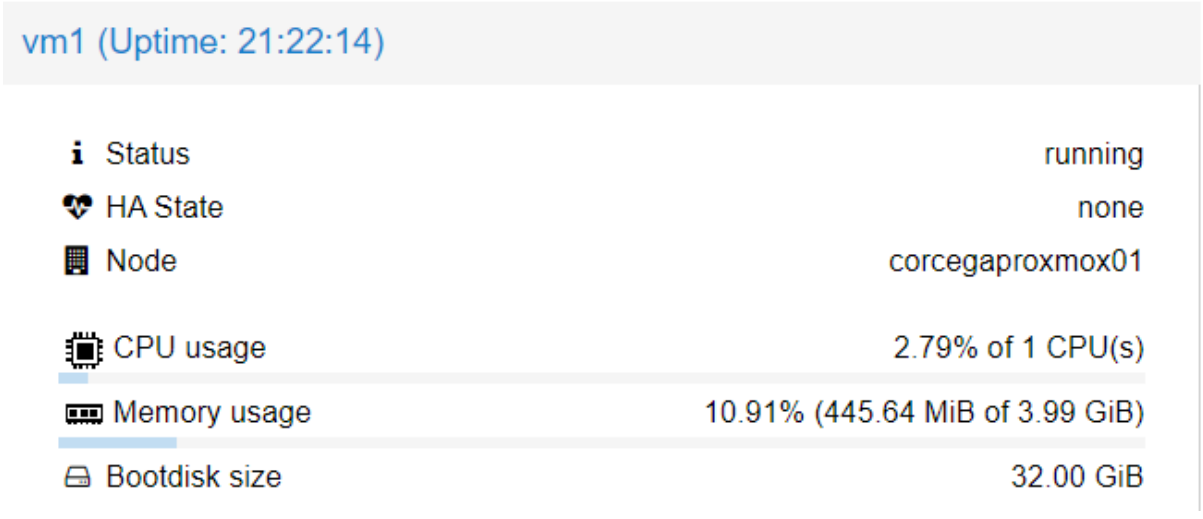
```
vm2@cloud:~$ ping 10.30.212.50
PING 10.30.212.50 (10.30.212.50) 56(84) bytes of data.
64 bytes from 10.30.212.50: icmp_seq=1 ttl=64 time=3.13 ms
64 bytes from 10.30.212.50: icmp_seq=2 ttl=64 time=0.650 ms
64 bytes from 10.30.212.50: icmp_seq=3 ttl=64 time=3.92 ms
64 bytes from 10.30.212.50: icmp_seq=4 ttl=64 time=0.796 ms
^C
--- 10.30.212.50 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.650/2.125/3.920/1.430 ms
```

Ping de la máquina vm2 a la vm1.

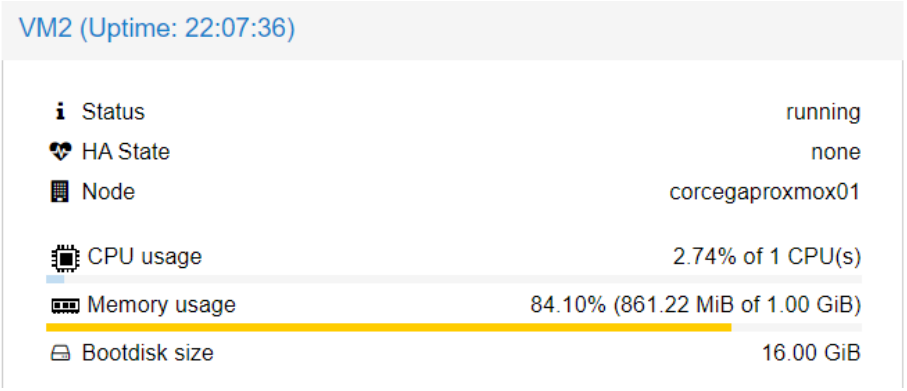
```
vm@cloud:~$ ping 10.30.212.41
PING 10.30.212.41 (10.30.212.41) 56(84) bytes of data.
64 bytes from 10.30.212.41: icmp_seq=1 ttl=64 time=1.16 ms
64 bytes from 10.30.212.41: icmp_seq=2 ttl=64 time=0.638 ms
64 bytes from 10.30.212.41: icmp_seq=3 ttl=64 time=0.646 ms
64 bytes from 10.30.212.41: icmp_seq=4 ttl=64 time=0.586 ms
^C
--- 10.30.212.41 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.586/0.758/1.164/0.235 ms
```

4. Gestión de Recursos

Aquí vemos el resumen de uso de la vm1.

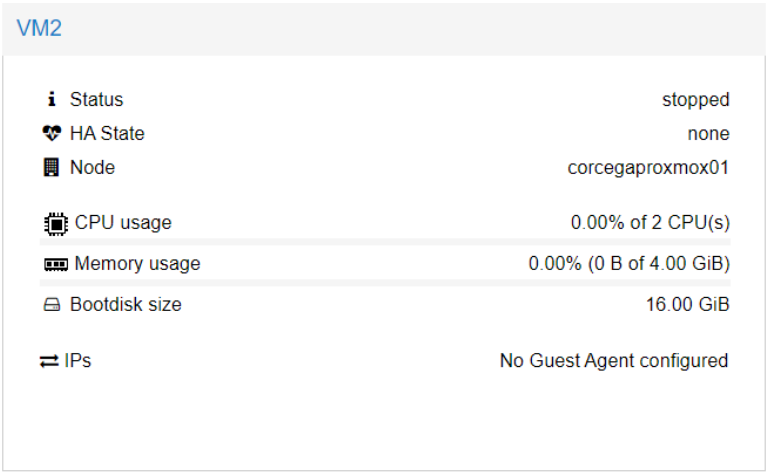


Y aquí vemos el resumen de uso de la vm2.

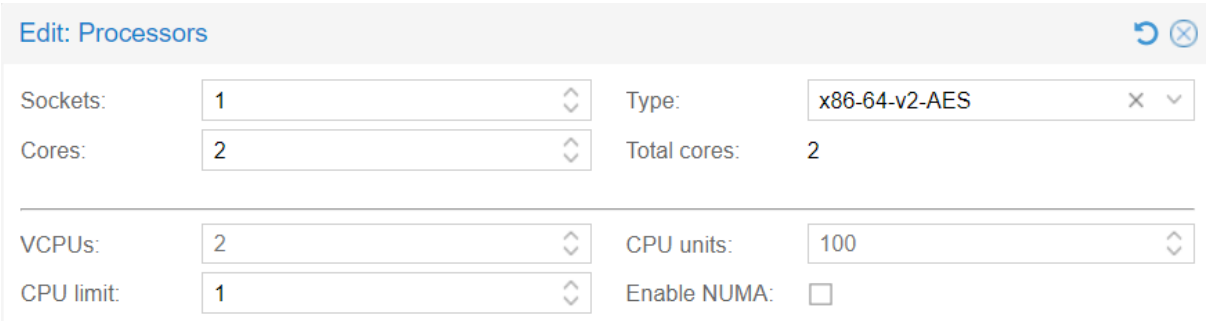


Ninguna está consumiendo más recursos de los esperados.

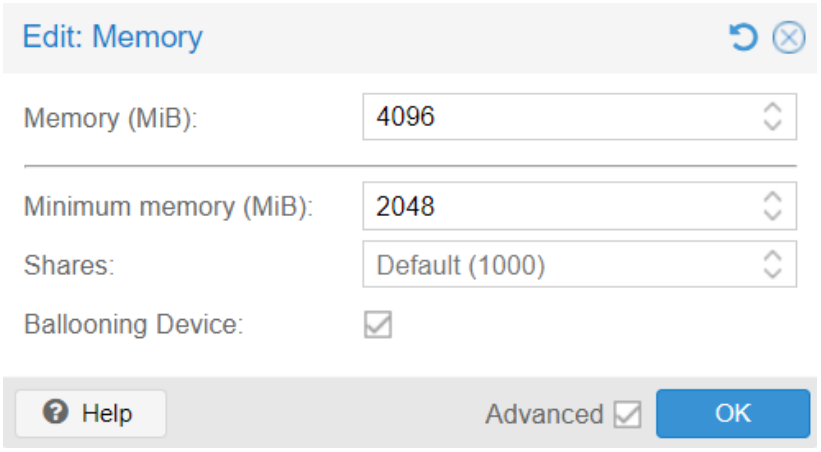
Si eso ocurriera lo arreglariamos de la siguiente manera:



Limitamos la CPU desde el modo avanzado en hardware.

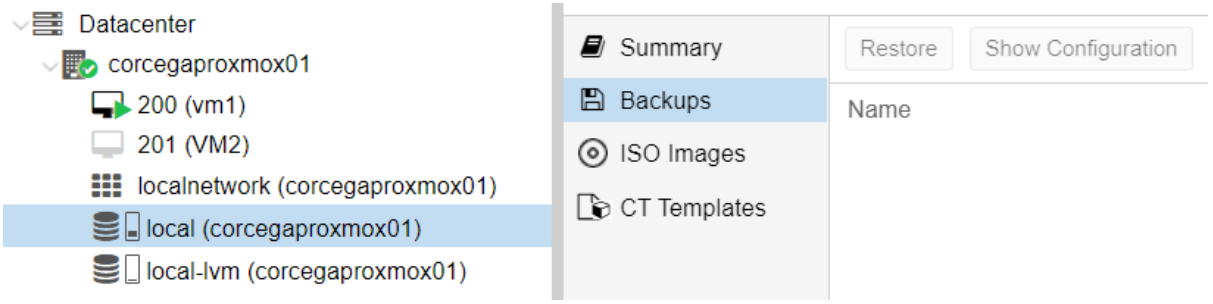


Limitamos también la memoria seleccionando el modo avanzado la bajamos a 2048 MiB.



5. Realización de Copias de Seguridad

Aquí es donde podemos ver los backups realizados.



Hacemos una prueba de restauración usando el backup del paso anterior.